U.S. Department of Commerce
National Oceanic and Atmospheric Administration
National Weather Service
National Centers for Environmental Prediction
5200 Auth Road
Camp Springs, MD 20746-4304

Office Note 454

DIAMOND INTERPOLATION: A CLASS OF ACCURATE COMPACT-STENCIL GRID INTERPOLATION METHODS

R. James Purser[*]
Science Applications International Corp., Beltsville, Maryland

April 10, 2007

This is an unreviewed manuscript, primarily intended for informal exchange of information among the NCEP staff members

[*] email: jim.purser@noaa.gov

Abstract

Throughout the execution of a data assimilation algorithm, it is necessary to apply, with numerous repetitions, the operator of interpolation from the regular analysis grid to interior target points at the irregular observation locations, and the adjoints of these operators. In such a situation, and where a reasonably high level of numerical accuracy is required, it is sometimes worth the expenditure of a little extra effort in establishing interpolation stencils and their weights that achieve the formal level of required accuracy with significantly fewer stencil points than would be required in the more conventional case of an interpolation scheme achieving the same formal order of accuracy by employing a Cartesian product of basic one-dimensional interpolation operators. A family of so-called 'diamond interpolation' schemes provides a natural way to construct alternative compact stencils and weights for a quasi-uniform rectangular grid in two or three dimensions. These methods, which employ approximately 'diamond'-shaped stencils, allow both values and derivatives of the gridded quantity to be interpolated. If continuity and smoothness of the target value as a function of position is a requirement, these properties can also be accommodated by modifications of these methods.

The techniques for generating and applying these interpolation schemes will be described here together with a brief discussion of some applications in numerical weather prediction.

1. Introduction

Many tasks in numerical weather prediction require interpolations, either from one regular grid to another, or from a regular grid to irregularly scattered points within the region covered by the grid. The grid of values upon which the interpolation operator acts is referred to as the 'source' grid, and a point at which an interpolation provides an approximate value based on the nearby surrounding source points is referred to as a 'target' point.

One typical application of interpolation in numerical weather prediction is the provision of initial conditions of a 'target' regional model from the gridded 'source' values supplied by a global model. Here, there is a high degree of regularity in both the source and target locations, which can sometimes be exploited by interpolating through an intermediate grid formed by intersections of one of the family of coordinates from the source grid with one of the family of coordinates of the target grid, provided the resulting intersections are transversal (i.e., they do not becomes tangential anywhere). In this case, one can seldom do better, in terms of computational efficiency, than applying suitable one-dimensional interpolations from the original source grid to the intermediate grid in a 'step one', followed by another set of one-dimensional interpolations from this intermediate grid to the final target grid in a 'step two'. Among the suitable one-dimensional interpolation techniques are: Lagrange polynomials; smoothed Lagrange polynomials (in order to guarantee continuity of derivatives with respect to target points); Hermite interpolation (when derivatives at source grid points are available); and polynomial spline methods. This form of 'dimensional splitting' of the grid-to-grid interpolation problem has been formalized in semi-Lagrangian models, where the process needs to be carried out each time step, and is referred to as the 'cascade method' (Purser and Leslie 1991, Leslie and Purser 1995.) It has been shown to improve the computational efficiency of interpolations in such models without compromising accuracy (e.g., Rančić 1995; Nair et al. 1999; Zerroukat et al. 2002).

Another typical situation occurs where *no* obvious regularity of the distribution of target points can be exploited. (An apt example is the case of data assimilation, where the targets are the observation points, which are essentially random.) The obvious and standard method of interpolating from a regular grid to an essentially random location inside the spanned region is to use a Cartesian product of the appropriate one-dimensional interpolation operators. Thus, for the example of a two-dimensional grid inscribed in Cartesian x and y directions, a bicubic interpolation is effected by interpolating four times in the x direction along the lines of contant y that straddle the target, then interpolating through the four intermediate target results in the y direction to attain the final target value (see Fig. 1). This requires that $4 \times 4 = 16$ original source values and four intermediate results are invoked to compute the target. In a tri-cubic interpolation in three dimensions, a total of $4 \times 4 \times 4 = 64$ original source values and 20 intermediate results are involved in the computation of the final target. However, if it is anticipated that several interpolations to the same target are going to be carried out, the 4×4, or $4 \times 4 \times 4$ stencil of weights can be precomputed to keep the computational cost of each application (in terms of multiplications and additions) equal to only 4^2, or 4^3, respectively. Formally, a 'fourth-order' of accuracy is achieved by such an interpolation using cubics, since the magnitude of the largest interpolation errors for the interpolation of a smooth source field scales in proportion to the fourth power of the source grid spacing. With centered stencils possessing a given even number, N, of points in each dimension, the interpolation formulae based on Lagrange polynomials have an order of accuracy equal to this N, an average operation count for the application of these scheme in two and three dimensions of approximately N^2 and N^3, respectively, and produce interpolated values that are continuous across grid box boundaries, but with discontinuities in their derivatives there.

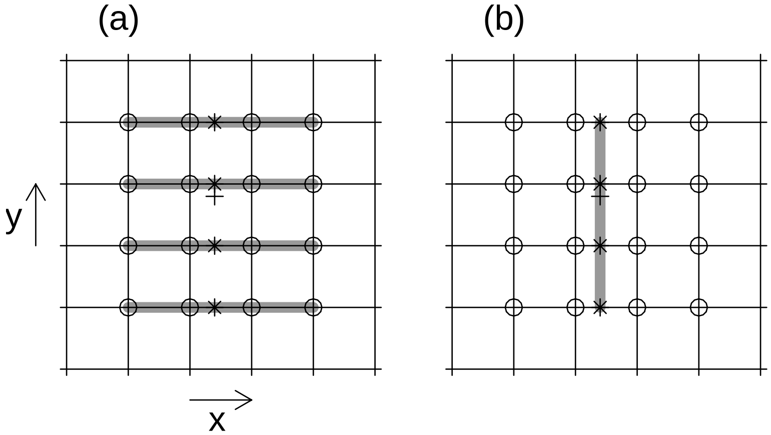

Figure 1. Schematic depiction of the conventional two-dimensional interpolation from a grid to a target point using a Cartesian product of one-dimensional interpolations. (a) Four interpolations in the x direction along grid lines of constant y; (b) Single interpolation in the y direction from the intermediate targets to the final target.

For about the same computational cost of application, it is possible to construct interpolations of a formal accuracy diminished by only one order, but possessing first derivatives of target values that now remain continuous. This is done by linearly-weighting, in each one-dimensional central interval, the pair of off-centered Lagrange polynomial formulae having an odd number, $N-1$, of points, which respectively omit each end point of the full one-dimensional centered stencil of the N points. It is because these schemes naturally break the problem into the separate dimensions that they are so convenient to apply and their weights are very efficient to calculate. Thus, they can be used also in grid-to-grid interpolations where, for geometrical reasons (such as the presence of polar singularities), the dimensional splitting options are not available. However, inherent in the Cartesian product construction of such multidimensional interpolation formulae is an unnecessarily accurate computation of the polynomial 'cross-terms' of a degree higher than the $N-1$ sufficient to achieve the formal Nth-order of accuracy. As a consequence, there exist other, smaller stencils than the Cartesian product stencils used for the multi-Lagrange polynomial interpolations, but which nevertheless attain a comparable accuracy. The generic family of 'diamond interpolations', which we shall develop in the later sections, exploits this opportunity to achieve an equal order of interpolation accuracy at a smaller cost of each application. The cost of computing the interpolation weights for the new methods is generally greater, but in cases where the same stencil is used repetitively, such as in data assimilation, or in cases of interpolation to model boundary and initial conditions, where geometrical consideration preclude dimensional splitting methods, the diamond interpolation formulae can be computationally advantageous.

2. Polynomial interpolation

In one dimension, the standard polynomial interpolation technique for the estimation $\bar{A}$ of a the value at a target straddled by an even number N of stencil points, $\{x_i\}, \quad i = 0, \ldots, N-1$, where the gridded values are $\{A_i\}$, is the Lagrange polynomial:

$$\bar{A}(x) = \sum_{i=0}^{N-1} W_i(x) A_i, \tag{2.1}$$

where the interpolation weights, W_i are defined:

$$W_i = \prod_{j \neq i} \frac{(x - x_j)}{(x_i - x_j)}. \tag{2.2}$$

An equivalent construction of the weights uses a Vandermonde matrix (Johnson and Riess 1982) to express the condition that a polynomial of the required degree fits the values at all the stencil's source points. The Vandermonde matrix $\boldsymbol{V}$ is the square $N \times N$ matrix of rows and columns numbered $0, 1, \ldots, N-1$, such that, with respect to a suitable local coordinate system, the general element of $\boldsymbol{V}$ is simply:

$$V_{i,j} = x_i^j. \tag{2.3}$$

A 'Vandermonde vector', $\boldsymbol{v}$, corresponding to the target point, x, can also be defined in the same way:

$$v_j = x^j, \tag{2.4}$$

and then, if the first N coefficients of the interpolating polynomial form the vector, $\boldsymbol{a} = [a_0, \dots, a_{N-1}]^T$, and the stencil values are similarly gathered into the vector, $\boldsymbol{A}$, we can use the collocation condition,

$$\boldsymbol{V}\boldsymbol{a} = \boldsymbol{A}, \tag{2.5}$$

to infer the polynomial coefficients $\boldsymbol{a}$, then apply them at the target to evaluate the interpolated value:

$$\boldsymbol{v}^T \boldsymbol{\cdot} \boldsymbol{a} = \bar{A}. \tag{2.6}$$

Assume the matrix $\boldsymbol{V}$ is nonsingular. Then, from (2.5) we have

$$\boldsymbol{v}^T \boldsymbol{\cdot} \boldsymbol{a} = \boldsymbol{v}^T \boldsymbol{V}^{-1} \boldsymbol{A}. \tag{2.7}$$

Thus,

$$\bar{A} = \boldsymbol{W}^T \boldsymbol{\cdot} \boldsymbol{A}. \tag{2.8}$$

where

$$\boldsymbol{W} = \boldsymbol{V}^{-T} \boldsymbol{v}. \tag{2.9}$$

One advantage of the Vandermonde representation of this problem is that, if it is not only the value, but also derivatives of the field, A, that we want to estimate at x, then these desired target values are obtained very conveniently from their Taylor series in terms of the coefficients, $\boldsymbol{a}$. In extending this method to higher dimensions, we find that another significant advantage is that the matrix method allows us to 'pick and choose' the stencil points and the fitted polynomial coefficients to some extent. It is this latter property that we shall exploit in the construction of more compact stencils in two or three dimensions.

3. The primary diamond interpolation scheme

Suppose that, in two dimensions, we wish to minimize the stencil sufficient to evaluate uniquely all the coefficients of the general interpolating polynomial of degree, $N-1$. The number of such coefficients is $M_P = N(N+1)/2$ and, just as these coefficients might most naturally be tabulated in a triangular array, a suggestive stencil of the same number of points might also take a triangular form, as illustrated for the case, $N = 4$, as the array of $M_P = 10$ circled points shown in Fig. 2a.

We shall show, by the following simple argument that, provided the labeled lines are distinct, this configuration does indeed provide sufficient information to determine all ten of the polynomial coefficients up to total degree, $N-1$, for a general polynomial in (x, y). First, we note that, for each term of total degree, $N-1$, in the polynomial, there is exactly one rectangular sub-array of the stencil whose dimensions in the x and y directions suggest the unique evaluation of the corresponding $(N-1)$-th-order derivative term by finite derivatives of the lowest order. For example, Fig. 2b shows, by the crosses, the six points of the stencil needed to evaluate the coefficient $a_{2,1}$ of the term, $a_{2,1}x^2y^1$, (of total degree, three) of the polynomial. Having obtained the four such degree-3 coefficients, i.e., $a_{3,0}$, $a_{2,1}$, $a_{1,2}$ and $a_{0,3}$, and subtracted their contributions from the original polynomial, we are left with a new residual polynomial of degree, two, and can dispense with the right-most points of the original stencil, i.e., the four points, (x_3, y_0), (x_2, y_1), (x_1, y_2), (x_0, y_3). The residual configuration of six stencil points

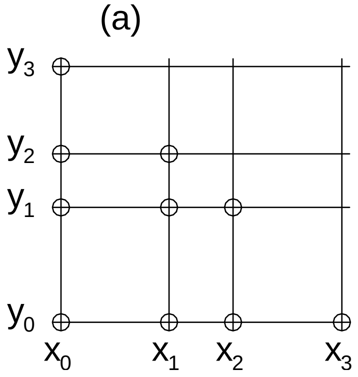

Figure 2. (a) A triangular array of points (circles) on a possibly nonuniform grid that will suffice to allow interpolation by a polynomial of total degree three. (b) the subset of that array (large crosses) from which it is possible to estimate the particular polynomial coefficient, $a_{2,1}$ (see text) of total degree three by finite differences.

suggests, in turn, the way to evaluate uniquely the three polynomial coefficients of total degree, two. Continuing in this way, we exhaust all the stencil points and accumulate all $M_P = 10$ of the complete polynomial coefficients.

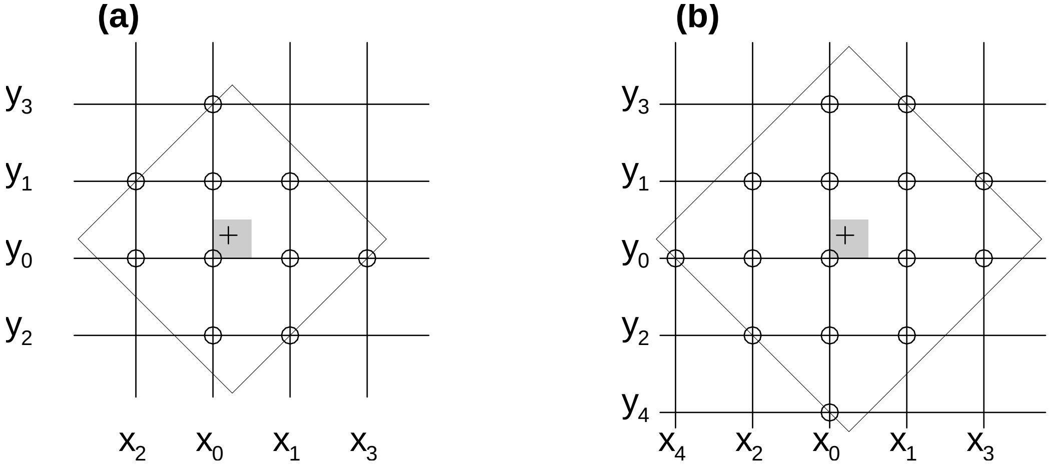

Figure 3. (a) A rearranged triangular array of points (circles), logically equivalent to that of Fig. 2, for a scheme with $N = 4$, but built upon lines that alternate in an expanding pattern on either side of the target point. The resulting 'diamond' stencil is more compactly arranged. (b) The diamond stencil for $N = 5$.

Note that the foregoing argument applies whether or not the lines of the stencil are uniformly spaced; Fig. 2 shows the lines as *un*equally spaced but, in fact, they need not even be arranged in monotonic order. The idea behind the basic construction of a diamond interpolation stencil is to exploit this freedom to choose the successive labeled lines in the different grid dimensions so that they fill out an alternating pattern moving outwards on either side of the target point, starting from whichever line is closest (in each separate dimension) to this target. Thus, in the case, $N = 4$ (but now depicting the source grid as a uniform one), the stencil of diamond interpolation points for the target (shown as the 'plus' sign) is the set of circled points of Fig. 3 that lie within the diamond-shaped region. Note that each labeled horizontal and vertical line of the stencil of Fig. 3 shares exactly the same set of transversally-labeled points as in the corresponding line of the stencil of Fig. 2. The algebraic structure of the calculation of stencil weights is therefore not essentially different. But the lines have been shuffled into a different order that makes the resulting stencil more compact. In terms of the L_1-norm, this construction guarantees the *most* compact stencil that suffices to allow interpolation of a given degree, so it is likely to be a robust and reliable choice numerically. If we define the L_1-norm distance for an assumed unit-spaced source grid to be,

$$L_1 = |x - x'| + |y - y'|, \tag{3.1}$$

then the 'radius', by this measure, of the diamond stencil centered, $(x_0 + 1/4, y_0 + 1/4)$, i.e., the center of the half-unit-square grid-box quadrant that contains the target point, is just $L_1 = N/2$. The generalization to an arbitrary number of grid dimensions is not hard to find; for example, in three dimensions, the number of stencil points to give an interpolation of formal order N is $M = N(N+1)(N+2)/6$, and the stencil points now lie inside a compact octahedral region of 'radius' $L_1 = (2N+1)/4$.

Although the stencil of the simplest diamond interpolation scheme always fits within a 'diamond-shaped' region (in the sense appropriate to the dimensionality involved), the fit is not a tight one along each side; that is, the stencil is slightly asymmetrical under mirror reflection in each of the coordinate directions, as we can immediately see in the examples of Fig. 3. Optimal use of the method requires that we not only identify the nearest grid point to the target to fix the lines labeled x_0, y_0, etc., but we must also identify the quadrant (in two dimensions) or octant (in three dimensions) adjoining this nearest grid point which contains the target. These regions are shown in gray in Fig. 3. Only then can we properly orient the diamond stencil to most compactly and centrally envelop the target.

Implementation of the basic diamond interpolation proceeds in two stages. First, upon identifying the location and orientation of the stencil, the weights, $\boldsymbol{W}_P$, are found by a linear inversion analogous to (2.9), but where the matrix, $\boldsymbol{V}$, is the appropriate two-dimensional or three-dimensional generalization of the Vandermonde matrix containing all the distinct polynomial terms up to total degree, $N - 1$, in the coordinates measured in grid units from the local stencil origin, (x_0, y_0) or (x_0, y_0, z_0). Note that if we adopt the convention that the point (x_0, y_0) serves as the local coordinate origin and we appropriately mirror-reflect (in either or both of the coordinate directions), then the matrix, $\boldsymbol{V}$, for a given N is the same for every stencil and therefore need only be inverted once.

The relative addresses of all the stencil points relative to this origin form a standard pattern common to all interpolations of a given order except for the small number (four, in two dimensions; eight, in three dimensions) of possible orientations corresponding to the mirror reflections in each coordinate direction. These orientations can be recorded as a simple 'corner code'. The second step is the implementation according to (2.8), but with the stencil positioned and oriented by the nominal center, (x_0, y_0), and corner code. To make the diamond interpolation approach worth the extra effort of computing its weights, we expect that the application step is one that will typically be performed at each target point (but with different gridded data, of course) a large number of times.

4. Refinements

(*a*) *Interpolating derivatives*

One simple refinement of the basic diamond interpolation method allows it to be used to evaluate the derivatives of the gridded quantity at the target point. For example, in two dimensions, where the vector $\boldsymbol{v}$ contains the various mixed powers of x and y:

$$v_{j(i_1,i_2)} = x^{i_1} y^{i_2}, \quad i_1 + i_2 < N, \tag{4.1}$$

and hence

$$\frac{\partial v_{j(i_1,i_2)}}{\partial x} = i_1 x^{i_1-1} y^{i_2}, \tag{4.2a}$$

$$\frac{\partial v_{j(i_1,i_2)}}{\partial y} = i_2 x^{i_1} y^{i_2-1}. \tag{4.2b}$$

Then the weights, $\boldsymbol{X}$ and $\boldsymbol{Y}$, defined by

$$\boldsymbol{X} = \boldsymbol{V}^{-T} \partial \boldsymbol{v} / \partial x, \tag{4.3a}$$

$$\boldsymbol{Y} = \boldsymbol{V}^{-T} \partial \boldsymbol{v} / \partial y, \tag{4.3b}$$

used with the same diamond stencil, produce the x- and y-partial derivatives of the gridded field at the target point, but to an order of accuracy now of only $N-1$. Higher degrees of derivatives are easily accommodated by extending this method in the obvious pattern.

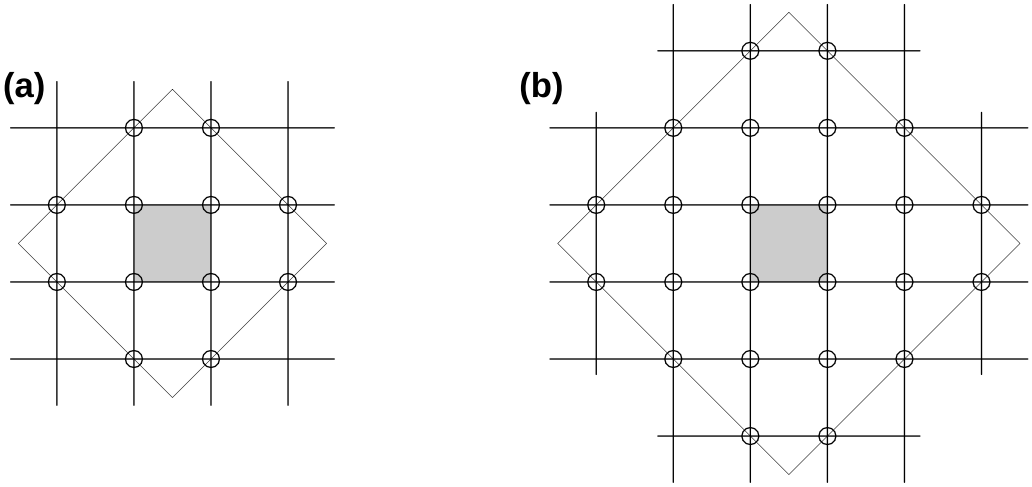

Figure 4. (a) Stencil (circles), of the symmetrized diamond interpolation scheme of order $N = 4$ and, (b) of order $N = 6$. The target point for these schemes lies within the central shaded grid box.

(*b*) *The 'S-diamond' scheme*

Using a basic polynomial interpolation scheme, we do not expect the derivatives at this target to remain continuous with respect to motion of the target point across the boundary of the domain of validity of each particular stencil. However, with diamond interpolation, the discontinuity is worse since it also affects the undifferentiated field itself. The discontinuities do not occur as the target moves from one grid box to the next, corresponding to a change in stencil orientation but not of stencil origin $\boldsymbol{x}_0$; instead, the discontinuities occur as the target crosses the center lines (or planes) of the grid box, corresponding to a change in the local origin ($\boldsymbol{x}_0$) *and* a change in stencil orientation. Thus, one remedy that removes the discontinuity of interpolated value is obtained by blending, bilinearly or trilinearly across the grid box, the four or eight basic diamond interpolation schemes nominally associated with all targets points in the neighborhood of the center of the grid box (where all the transitional center lines, or planes, intersect). For an even N, this scheme makes the stencil a bit larger, but also conveniently makes it a more symmetrical shape, as we can see for the two-dimensional examples shown, for $N = 4$ and $N = 6$, in Fig 4. (The attempt to form a scheme of this kind with odd N leads to exactly the same stencil as the even $N + 1$ scheme, so the odd schemes confer no numerical advantage.) Distinguishing these 'secondary modifications', or ('S-diamond') schemes with even N from the original 'primary' ('P-diamond') schemes, we find that the stencil sizes of the even-order S-schemes are given by:

$$M_S = 2N(2N + 2)/2, \tag{4.4}$$

in two dimensions, and:

$$M_S = 2N(2N + 2)(2N + 4)/6, \tag{4.5}$$

in three dimensions.

Note that, for the case of $N = 4$, the stencil pattern is exactly the same as that described in Simmons (1991), but attributed there to Phillipe Courtier, for applications in semi-Lagrangian models. However, it does not appear that the methods used in the semi-Lagrangian context achieve the formal order of accuracy, $N = 4$, that the more careful calculation of the stencil coefficients by the bilinearly-weighted symmetrized S-diamond method allows. Unfortunately, the symmetrized diamond stencil only partially addresses the problem of smoothness of target values; while undifferentiated values remain continuous, the method does not remove the discontinuities of the derivatives, which persist.

(*c*) *The 'T-diamond' scheme*

A more fulfilling remedy, which removes discontinuities in both the interpolated value *and* the interpolated derivatives, can be obtained by a slightly more general averaging of the basic diamond schemes, three in a row in each dimension, and weighted according to a profile based on quadratic 'B-splines' (as defined, for example, in Schoenberg 1967, de Boor 1978). B-splines are closely related to piecewise-polynomial maximally-smooth partitions of unity. For example, on a one-dimensional unit grid, define the positive remainder, $r = x - \mathsf{FLOOR}(x)$, where FLOOR denotes the largest integer not larger than x (as in the $\mathsf{FORTRAN}$ function of the same name). Then the functions,

$$S_i(r) = \begin{cases} 0 & : \quad i < 1 \\ r^2/2 & : \quad i = 1 \\ (1 + 2r - r^2)/2 & : \quad i = 2 \\ 1 & : \quad i > 2 \end{cases}, \tag{4.6}$$

serve as smooth partitions of unity in each unit segment of x, as illustrated in Fig. 5a. Therefore, by defining the quadratic ‘B-spline’ segements:

$$B_i(r) = S_i(r) - S_{i-1}(r), \tag{4.7}$$

the fundamental B-spline function can be assembled:

$$B(x) = \left\{ \begin{array}{lcllcccc} B_{<1}(r) & \equiv & 0 & : & & & x & < & 0 \\ B_1(r) & \equiv & r^2/2 & : & 0 & \leq & x & < & 1 \\ B_2(r) & \equiv & (1+2r-2r^2)/2 & : & 1 & \leq & x & < & 2 \\ B_3(r) & \equiv & (1-2r+r^2)/2 & : & 2 & \leq & x & < & 3 \\ B_{>3}(r) & \equiv & 0 & : & 3 & \leq & x & & \end{array} \right. . \tag{4.8}$$

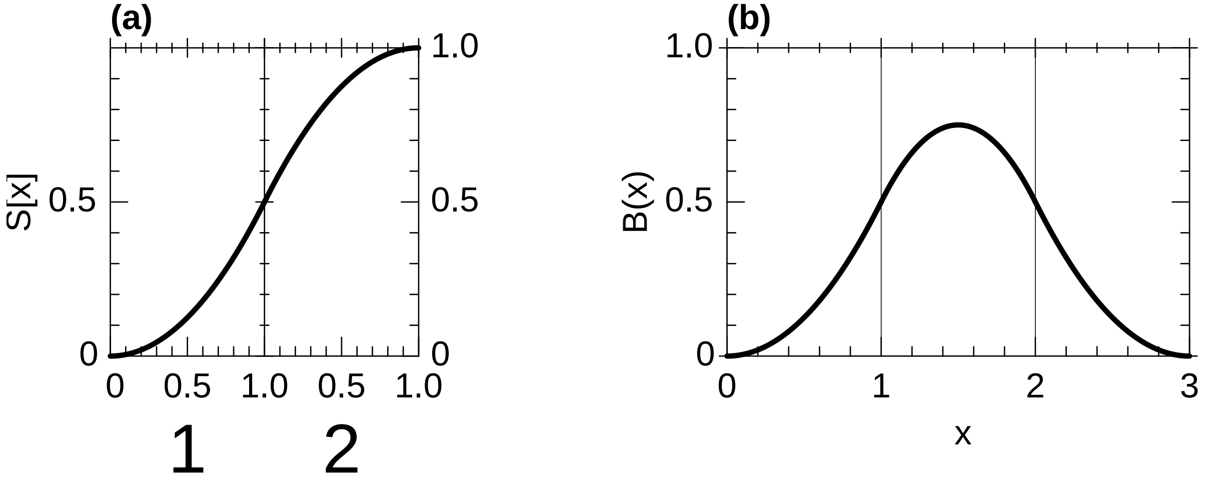

Figure 5. (a) Graphs of the two functions, $S_1(r)$ and $S_2(r)$ which smoothly partition unity, and from which the quadratic B-spline is constructed. (b) the graph of the fundamental quadratic B-spline for a regular unit grid.

The non-trivial portion of this function is shown in Fig. 5b. $B(x)$ is continuous in both its value and in its first derivative as x varies and, since

$$\sum_{i=-\infty}^{\infty} B(i+r) = 1, \quad \forall r \in [0,1), \tag{4.9}$$

it can be used to form a progressive and smooth weighted average of a sequence of terms, such as the successive polynomial interpolation scheme evaluations of a given target value. With this smooth average of such schemes, regardless of the lack of continuity between consecutive members, their aggregrated result will be guaranteed continuous in value and derivative. In the case of diamond interpolation in two dimensions of a given order, N, for a grid formed by the integer values of x and y, denote by ‘$\boldsymbol{W}_{P,(I,J)}(x,y)$’ the basic weights deemed appropriate for the target (x,y) in the grid box quadrant for which,

$$I = \mathsf{FLOOR}(2x), \tag{4.10a}$$

$$J = \mathsf{FLOOR}(2y). \tag{4.10b}$$

Note that I and J change at twice the rate as the grid indices and effectively index a double-resolution grid. It is also convenient to define the corresponding double-resolution remainders:

$$q_x = 2x - I, \tag{4.11a}$$

$$q_y = 2y - J. \tag{4.11b}$$

Then this B-spline-smoothed form of the diamond interpolation scheme, which we shall refer to as the 'tertiary modified', or 'T-diamond' scheme, and whose weights will be denoted $\boldsymbol{W}_{T,(I,J)}(x,y)$, can be defined as the a proper weighted average of the nearest 3×3 array of the basic P-diamond interpolations:

$$\boldsymbol{W}_{T,(I,J)}(x,y) = \sum_{J'=-1}^{1} \sum_{I'=-1}^{1} B(q_x + 1 - I')B(q_y + 1 - J')\boldsymbol{W}_{P,(I+I',J+J')}(x,y). \tag{4.12}$$

The weight given to any one of the contributing P-diamond schemes in (4.12) is seen to be a function of (x,y) formed as the Cartesian product of the x- and y-oriented B-splines (each with the profile shown in Fig. 5b) on a half-unit grid spanning the target, shown in Fig. 6 as the shaded region of nine half-unit squares. The contribution of the particular P-scheme, being a polynomial, is perfectly smooth; therefore the smoothness and continuity of the T-scheme is determined entirely by the smoothness and continuity of the B-spline weighting.

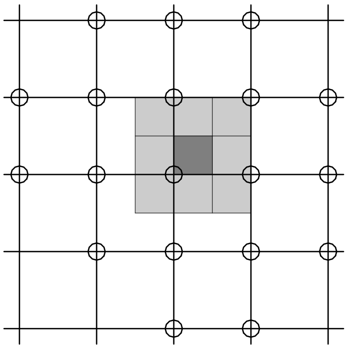

Figure 6. Stencil (circles) of the T-diamond scheme of order $N = 4$. The central set of nine small gray-shaded squares indicate the nominal centers of each of the P-diamond schemes that contribute, with B-spline weightings, to the definition of this T-diamond scheme, as described in the text. The central, darker shaded square contains the actual target

Being the superposition of nine of the original diamond stencils, the resulting stencil of $\boldsymbol{W}_T$ is not quite so obviously diamond-shaped. The case with $N = 4$ is illustrated in Fig. 6, where the target is contained within the darker gray square, and the principal target domains of the contributing P-diamond schemes are indicated by the surrounding 3×3 array of shaded half-unit squares. The complexity of its construction makes it computationally less attractive except in applications where smooth continuity of the interpolated values is important and where, as before, it is expected that multiple applications of the same set of weights will be required to make the investment involved in computing these weights worthwhile.

We can compare the results of each of these schemes for $N = 4$. Fig. 7 shows the interpolated results, at very high target resolution, along the same random transect through the interior of

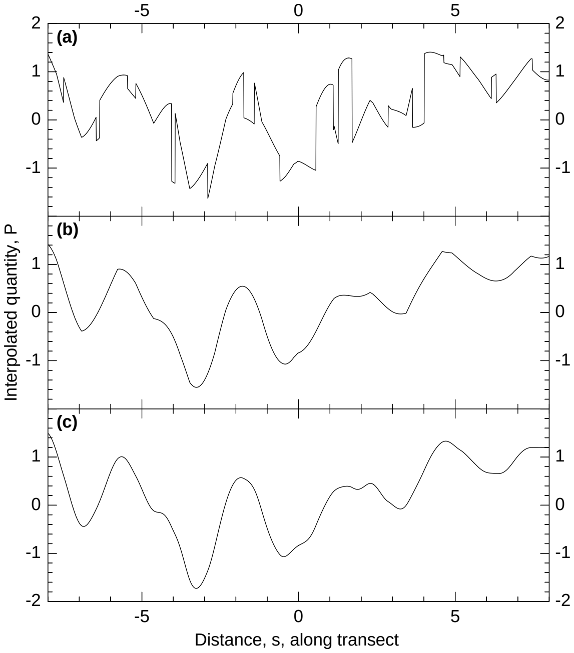

Figure 7. Interpolated values of a quantity, P interpolated at very high resolution along a randomly oriented transect through the interior of a three dimensional unit grid whose source values are initialized with independent unit-variance Gaussian random numbers. (a) The values obtained with the primary diamond scheme, 'P' at $N = 4$, showing that values are piecewise-polynomials without continuity wherever there is a transition in the stencil. (b) The 'S' scheme in which the stencil is slightly enlarged and symmetrized, showing that the interpolated value is now continuous along the transect, but possesses slight kinks at transitions of these stencils. (c) the results for the same transect and data using the 'T' diamond scheme with the same $N = 4$, showing a higher degree of smoothness.

a three-dimensional grid of source points where, in order to emphasize any tendency to form discontinuities, the test values on the grid have been given as completely independent unit-variance Gaussian random numbers. Panel (a) shows the 'P-scheme', and, as expected, the random gridded data are interpolated to piecewise continuous segments exhibiting some positive correlation, but no formal continuity, at the points of transition through the grid wherever the interpolation stencil of source points changes. Panel (b) shows the S-diamond scheme results for the same data and transect, where now the continuity of value, but not of the derivative, is clearly evident. Finally, the panel (c) shows the superior continuity and smoothness achieved by employing the B-spline-smoothed T-diamond scheme.

5. Discussion

Three families, 'P', 'S' and 'T', of diamond interpolation schemes have been defined and compared. They are intended for applications where the greater cost of obtaining the stencil weights (compared to conventional Cartesian product interpolators of the same order N) can be offset by the greater efficiency of the application of the schemes in situations where the applications to each given fixed target are repeated many times with only the gridded source data varying between applications. It is helpful to quantify this relative efficiency by tabulating the stencil size of each scheme compared to the nearest equivalent Cartesian-product scheme.

This is done in Table 1 for the two-dimensional schemes, and in Table 2 for the corresponding three-dimensional schemes. In two dimensions, the T-scheme stencil contains M_T points where,

$$M_T = [(N+1)^2 + 3(N+1) - 2]/2 \tag{5.1}$$

which is smaller than the corresponding Cartesian product stencil of bilinearly weighted Lagrange polynomials [requiring $(N+1)^2$ points] for $N > 1$. In three dimensions, the modified stencil contains

$$M_T = [(N+1)^3 + 9(N+1)^2 + 8(N+1) - 12]/6 \tag{5.2}$$

points, which is, again, less than the corresponding Cartesian product stencil's size of $(N+1)^3$ for any $N > 1$. Tabulations of these various stencil sizes in two and three dimensions are given in Tables 1 and 2 for N up to six.

The space-dependent weightings involved in the construction of the S- and T-diamond schemes complicates the evaluations of the derivatives at the target points, and no attempt has been made to carry this procedure beyond the evaluation of first derivatives. The basic P-diamond method is not subject to such complications and would therefore be the scheme of choice for applications where it desirable to construct a low-order Taylor series about a single target point. In the data assimilation context, examples where this could be of value is where a 'vertical' string of observations is in actuality a slanting vertical curve, such as the path of an ascending radiosonde, the oblique rays of many of the off-nadir satellite soundings, or the locations at which GPS radio occultation rays intersect the more substantial portions of the atmosphere (where the significant phase delays occur) at each occultation 'event'.

Another application of the described method is in the provision of boundary (or initial) conditions of a model derived by interpolation from another model which contains it. A variation of the theme involves the 'Yin-Yang' configuration (Kageyama and Sato 2004) of twin domains, mutually nested, where each obtains boundary conditions, at every significant step of the dynamics, from its counterpart. Since the geometrical configuration of one grid relative to the other does not change, the computation of the weights associated with diamond interpolation stencils can be done once and the efficiencies associated with the smaller stencils (for a given order of accuracy) provide a useful economy for the duration of the model's simulation.

Early tests of the diamond interpolation method considered the use of grid spaced nonuniformly in their coordinates. However, in addition to suffering from an increase of complexity, these methods tended to exhibit large excursions of the interpolated values between grid points in a behavior reminiscient of (but not exactly equivalent to) the 'Runge' effect (that degrades the value of significantly uncentered high-order polynomial interpolations even on uniform grids). Therefore, it is recommended that, in cases where the given grid is non-uniform, a coordinate transformation that forces the grid spacings in each direction to become uniform in the new coordinate be adopted prior to application of the diamond schemes.

A package of FORTRAN-90 utility codes that facilitate the construction and application of the methods described in this note for unit-spaced grids has been written (and is available, upon request, from the author). Methods of the 'diamond' type have already been used for several years at NCEP in the data assimilation context and are now also being considered for implementation in the boundary-reconciliation procedure in investigations of model configurations employing the 'Yin-Yang' grid framework.

TABLE 1. Comparison of stencil sizes in two dimensions

N	M_P	M_S	N^2	M_T	$(N+1)^2$
2	3	4	4	8	9
3	6		9	13	16
4	10	12	16	19	25
5	15		25	26	36
6	21	24	36	34	49

TABLE 2. Comparison of stencil sizes in three dimensions

N	M_P	M_S	N^3	M_T	$(N+1)^3$
2	4	8	8	20	27
3	10		27	38	64
4	20	32	64	63	125
5	35		125	96	216
6	56	80	216	138	343

Acknowledgment

The author is grateful to Dr. David F. Parrish, who enthusiastically advocated the original basic diamond interpolation schemes for the applications in data assimilation.

References

de Boor, C. 1978 *A Practical Guide to Splines.* Springer, New York. 392 pp.

Johnson, L. W., and R. D. Riess 1982 *Numerical Analysis (2nd Edition)*, Addison Wesley, Reading MA. 563 pp.

Kageyama, A., and T. Sato 2004 The "Yin-Yang grid": An overset grid in spherical coordinates. *oai:arXiv.org:physics/0403123*

Leslie, L. M., and R. J. Purser 1995 Three-dimensional mass-conserving semi-Lagrangian scheme employing forward trajectories. *Mon. Wea. Rev.,* **123,** 2551–2566.

Nair, R., J. Côté, and A. Staniforth 1999 Cascade interpolation for semi-Lagrangian advection over the sphere. *Quart. J. Roy. Meteor. Soc.,* **125,** 1445–1468.

Purser, R. J., and L. M. Leslie 1991 An efficient interpolation procedure for high-order three-dimensional semi-Lagrangian models. *Mon. Wea. Rev.,* **119,** 2492–2498.

Rančić, M. 1995 An efficient conservative, monotonic remapping as a semi-Lagrangian transport algorithm. *Mon. Wea. Rev.,* **123,** 1213–1217.

Schoenberg, I. J. 1967 On spline functions, in *Inequalities*, O. Shisha, ed., Academic Press, New York, pp. 255–291.

Simmons, A. J., 1991 Development of a high resolution semi Lagrangian version of the ECMWF forecast model. *ECMWF Technical Memorandum 180.* (Available from ECMWF, Shinfield Park, Reading, RG2 9AX, U.K.)

Zerroukat, M., N. Wood, and A. Staniforth 2002 SLICE: A semi-Lagrangian inherently conserving and efficient scheme for transport problems. *Quart. J. Roy. Meteor. Soc.,* **128,** 2801–2820.